Univerzitet u Zenici

Politehnički fakultet

Softversko inženjerstvo

Predmet: Razvoj mobilnih aplikacija

Godina studija: III (treća)

SEMINARSKI RAD

WEATHER APP

Profesor: Asmir Butković

Studenti: Nejra Čoloman 92

Adin Jahić

Datum: Januar, 2022.god.

SADRŽAJ

1. Uvod.....1

2. Početna stranica 2

3. Registracija..... 5

4. Login 8

5. Lokacija..... 10

6. Informacije o vremenu.....12

7. Zaključak..... 19

8. Popis slika..... 20

1. Uvod

Živimo u vremenu u kojem se klima mijenja i u skladu s tim vrijeme se mijenja iz dana u dan. Da bismo se mogli adekvatno pripremiti za svakodnevne obaveze potrebno nam je da znamo kakva je vremensa prognoza. U tome nam pomaže aplikacija za vremensku prognozu koja nam je lahko dostupna na našim mobilnim uređajima.

Vremenska prognoza kao aplikacija trebala bi omogućiti korisnicima prikaz informacija o današnjem vremenskom stanju specifičnom za uneseni grad. Informacije koje je prikazuju su: temparatura u stepenima celzija, vlažnost, brzina vjetra, oblačnost i pritisak. Također se nude informacije o vremenu za naredna čeiri dana.

U nastavku, služeći se slikama iz emulatora, kodovima i opisima, pokušat ćemo objasniti rad i funkcionalnosti aplikacije.

2. Početna stranica

Dizajn aplikacije je osmišljen tako da prilikom pokretanja iste, ulazimo na početnu stranicu odnosno, u aplikaciji nazvanu kao SplashScreen.xaml. Na početnoj stranici se nalaze dva funkcionalna buttona Login i Register.

Slika 1. Početna stranica aplikacije

Kod za pomenutu stranicu (xaml).

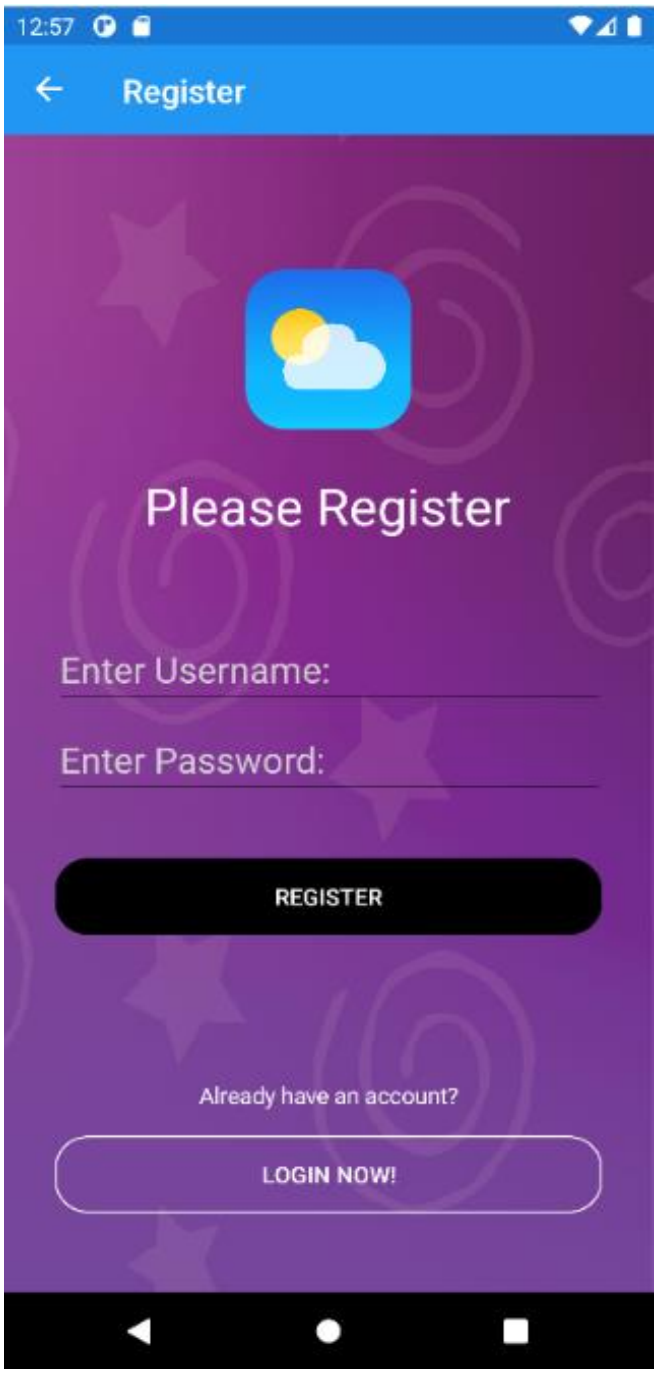
```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="ProjekatRMA.Views.SplashScreen"
             BackgroundImage ="background.jpg"
             Title="Homepage"
             >
    <ContentPage.Content>
        <StackLayout VerticalOptions="Center">
            <Frame BackgroundColor="MediumPurple" CornerRadius="20" Margin="20" HeightRequest="500" Opacity="0.9">
                <StackLayout VerticalOptions="CenterAndExpand">
                    <Image Source="Logo.png" WidthRequest="150" HeightRequest="150" HorizontalOptions="Center" VerticalOptions="Center"></Image>
                    <Label Text="Welcome to my Weather App" HorizontalOptions="Center" HorizontalTextAlignment="Center" FontSize="32" Margin="0,0,0,60" TextColor="Black"
                           FontAttributes="Bold"></Label>
                    <Button Clicked="RegisterBtn" Text="Login" Padding="20,10" HorizontalOptions="Center" WidthRequest="200" TextColor="Purple" BackgroundColor="Transparent"
                            BorderColor="Purple" BorderWidth="2" CornerRadius="20" ></Button>
                    <Button Clicked ="LoginBtn" Text="Register" Padding="20,10" HorizontalOptions="Center" WidthRequest="200" BackgroundColor="Black" TextColor="White"
                            CornerRadius="20"></Button>
                </StackLayout>
            </Frame>
        </StackLayout>
    </ContentPage.Content>
</ContentPage>
```

Pozadinski kod u C#

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using Xamarin.Forms;
using Xamarin.Forms.Xaml;

namespace ProjekatRMA.Views
{
    [XamlCompilation(XamlCompilationOptions.Compile)]
    public partial class SplashScreen : ContentPage
    {
        public SplashScreen()
        {
            InitializeComponent();
        }

        private void RegisterBtn(object sender, EventArgs e)
        {
            Application.Current.MainPage.Navigation.PushAsync(new Login());

        }

        private void LoginBtn(object sender, EventArgs e)
        {
            Application.Current.MainPage.Navigation.PushAsync(new Register());

        }
    }
}
```

3. Registracija

Klik na dugme Register nas vodi na stranicu Register.xamal. Na ovoj stranici se nalaze dva polja za unos username-a i lozinke, te dva buttona Register i Login now. Nakon unosa podataka i klika na button Register podaci se spremaju u InMemoryDB simulirana statičkom klasom sa statičkim članovima. Ako već imamo račun, drugi button, Login now nam daje mogućnost da se ulogujemo na pomenuti račun tako što nas preusmjerava na Login.xamal stranicu.



Slika 2. Stranica za registraciju

Kod za Register.xaml stranicu

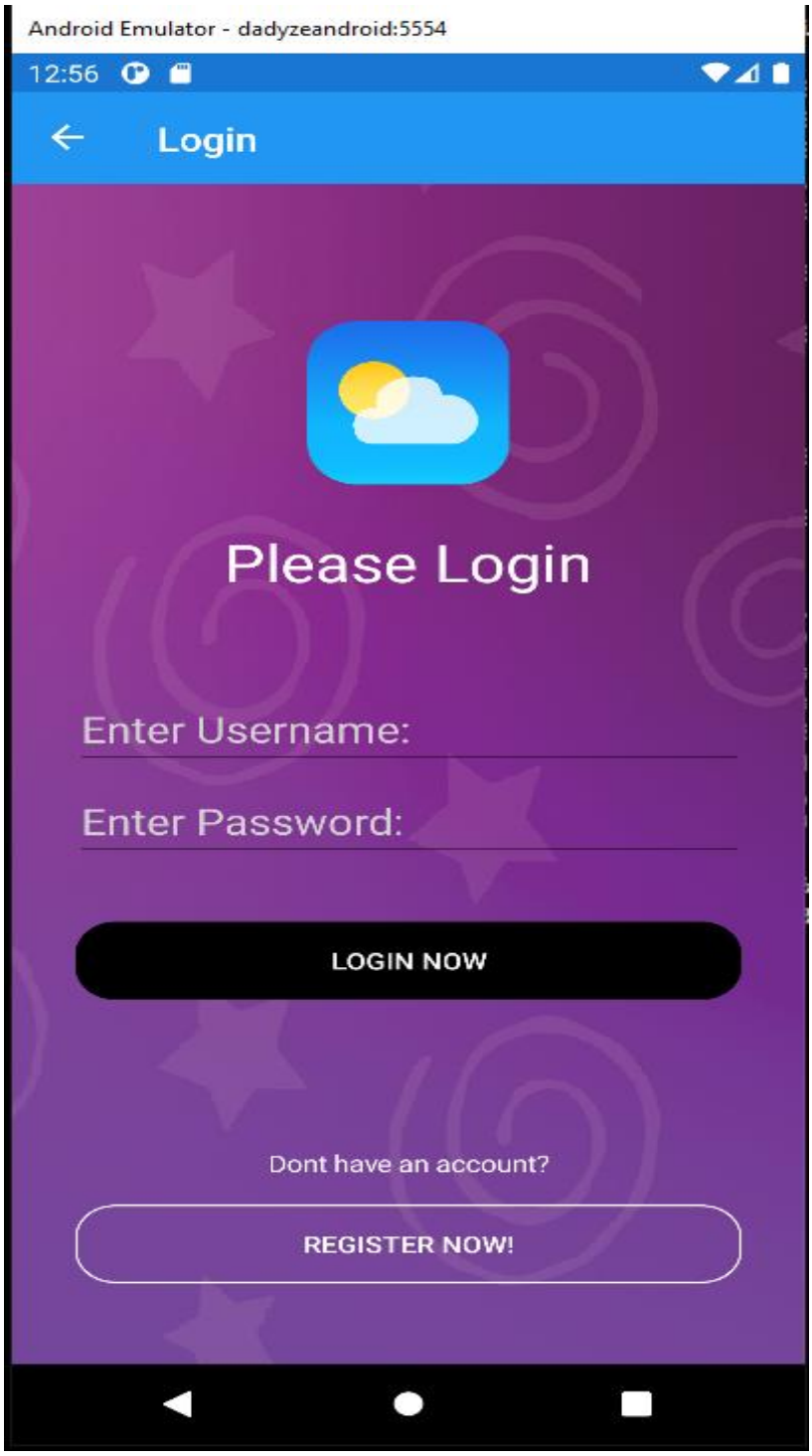
```
<?xml version="1.0" encoding="utf-8"?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="ProjekatRMA.Views.Register"
             BackgroundImage ="background.jpg"
             Title="Register"
             >



    <ContentPage.Content>
        <StackLayout>
            <StackLayout VerticalOptions="CenterAndExpand">
                <Image Source="Logo.png" HeightRequest="100" WidthRequest="100"></Image>
                <Label Text="Please Register" FontSize="32" Padding="20" HorizontalOptions="CenterAndExpand" HorizontalTextAlignment="Center"
                  TextColor="White"></Label>
                <StackLayout Padding="30" >
                    <Entry x:Name="username" Placeholder="Enter Username: " TextColor="White" PlaceholderColor="LightGray" FontSize="Large"></Entry>
                    <Entry x:Name="pass" Placeholder="Enter Password: " TextColor="White" PlaceholderColor="LightGray" FontSize="Large" IsPassword="True"></
                      Entry>
                </StackLayout>
                <Button Clicked ="RegisterBtn" Text="Register" Padding="20,10" BackgroundColor="Black" TextColor="White" BorderColor="White"
                        WidthRequest="330" HorizontalOptions="Center" CornerRadius="20"></Button>
            </StackLayout>


            <Label Text="Already have an account?" FontSize="Small" TextColor="White" HorizontalTextAlignment="Center"></Label>
            <Button  Clicked="RegisterBtnRediretLogin" TextColor="white" Text="Login now!" Padding="20,10"  WidthRequest="330"
                    HorizontalOptions="Center" Margin="0,10,0,50" BorderColor="White" BorderWidth="1" BackgroundColor="Transparent" CornerRadius="20"></Button>
        </StackLayout>
    </ContentPage.Content>
</ContentPage>
```

Pozadinski kod u C#

```
using ProjekatRMA.Models;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using Xamarin.Forms;
using Xamarin.Forms.Xaml;

namespace ProjekatRMA.Views
{
    [XamlCompilation(XamlCompilationOptions.Compile)]
    public partial class Register : ContentPage
    {
        public Register()
        {
            InitializeComponent();
        }

        private void RegisterBtn(object sender, EventArgs e)
        {
            DisplayAlert("Msg", "User Created Successfully", "ok");
            UserData.Users.Add(new UserObj(username.Text, pass.Text));
            Application.Current.MainPage.Navigation.PushAsync(new Login());
        }

        private void RegisterBtnRediretLogin(object sender, EventArgs e)
        {
            Application.Current.MainPage.Navigation.PushAsync(new Login());
        }
    }
}
```

4. Login

Klik na button Login na početnoj stranici nas vodi na Login.xamal stranicu. Na ovoj stranici se nalaze dva polja za unos username-a i lozinke, te dva buttona Login now i Register now. Nakon unosa podataka i klika na button Login now radi se LINQ na InMemoryDB koji traži da li submitovani user postoji. Ako postoji šalje na sranicu WeatherLocation i šalje usera kao konstruktor za održavanje sesije. Ako ne postoji izbacuje error.



Slika 3. Login stranica

Kod za Login.xaml

```xaml
<?xml version="1.0" encoding="utf-8"?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="ProjekatRMA.Views.Login"
             BackgroundImage ="background.jpg"
             Title="Login"
             >


    <ContentPage.Content>
        <StackLayout>
            <StackLayout VerticalOptions="CenterAndExpand">
                <Image Source="Logo.png" HeightRequest="100" WidthRequest="100"></Image>
                <Label Text="Please Login" FontSize="32" Padding="20" HorizontalOptions="CenterAndExpand" HorizontalTextAlignment="Center" TextColor="White"></
                  Label>
                <StackLayout Padding="30" >
                    <Entry x:Name="username" Placeholder="Enter Username: " TextColor="White" PlaceholderColor="LightGray" FontSize="Large"></Entry>
                    <Entry x:Name="pass" Placeholder="Enter Password: " TextColor="White" PlaceholderColor="LightGray" FontSize="Large" IsPassword="True"></
                      Entry>
                </StackLayout>
                <Button Clicked ="LoginSubmitBtn" Text="Login now" Padding="20,10" BackgroundColor="Black" TextColor="White" BorderColor="White"
                  WidthRequest="330" HorizontalOptions="Center" CornerRadius="20"></Button>
            </StackLayout>
            <Label Text="Dont have an account?" FontSize="Small" TextColor="White" HorizontalTextAlignment="Center"></Label>
            <Button  Clicked="LoginBtnRediretRegister" TextColor="white" Text="Register now!" Padding="20,10"  WidthRequest="330" HorizontalOptions="Center"
              Margin="0,10,0,50" BorderColor="White" BorderWidth="1" BackgroundColor="Transparent" CornerRadius="20"></Button>
        </StackLayout>
    </ContentPage.Content>
</ContentPage>
```

Kod u C#

```csharp
using ProjekatRMA.Models;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using Xamarin.Forms;
using Xamarin.Forms.Xaml;

namespace ProjekatRMA.Views
{
    [XamlCompilation(XamlCompilationOptions.Compile)]
    public partial class Login : ContentPage
    {
        public Login()
        {
            InitializeComponent();
        }

        private void LoginSubmitBtn(object sender, EventArgs e)
        {
            if (UserData.Users.Any(x => x.Name == username.Text && x.Password == pass.Text))
            {
                UserObj DTO = UserData.Users.SingleOrDefault(x => x.Name == username.Text && x.Password == pass.Text);
                Application.Current.MainPage.Navigation.PushAsync(new WeatherLocation(DTO));
            }
        }

        private void LoginBtnRediretRegister(object sender, EventArgs e)
        {
            Application.Current.MainPage.Navigation.PushAsync(new Register());
        }
    }
```

5. Lokacija

Nakon što se korisnik uloguje aplikacija ga vodi na WeatherLocation.xamal stranicu. Na pomenutoj stranici se nalazi poruka dobrodošlice i ispisuje se username. Ispod toga se nalazi polje za unos lokacije (grada) za koju želimo provjeriti vremensku prognozu. Na stranici se nalazi i button Show weather koji nas šalje na stranicu WeatherApp.

Slika 4. Stranica za unos lokacije

Kod za WeatherLocation.xaml

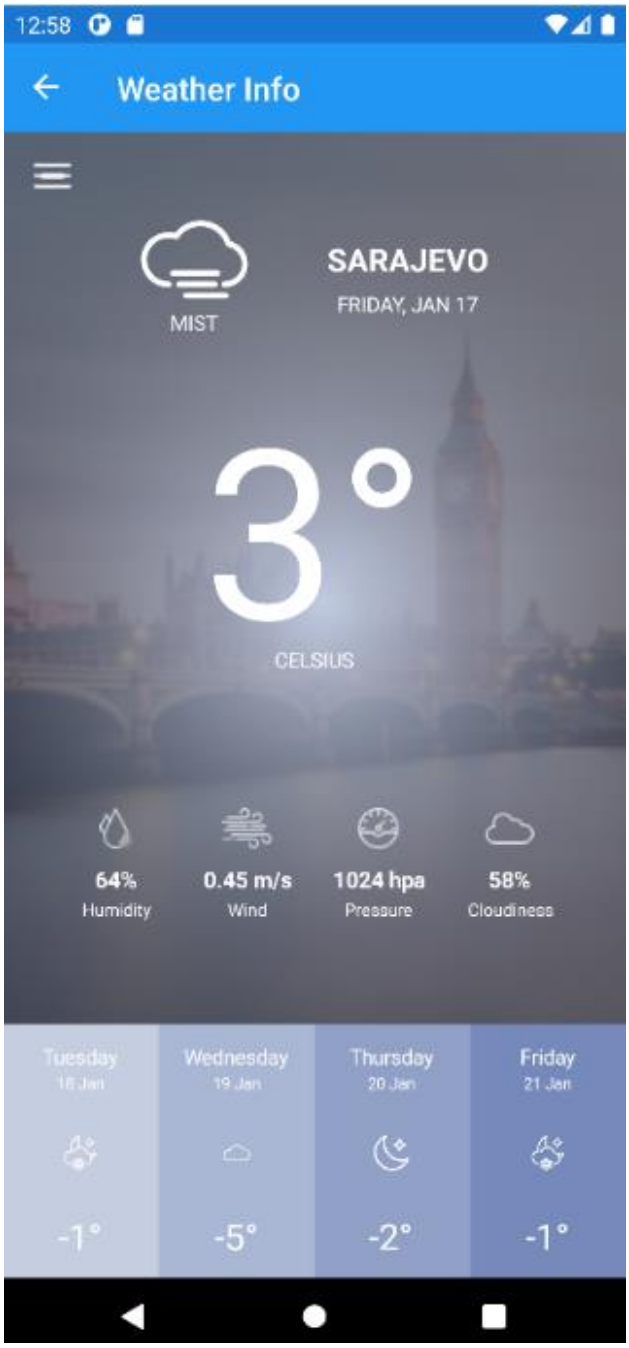
```xaml
<?xml version="1.0" encoding="utf-8"?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="ProjekatRMA.Views.WeatherLocation"
             BackgroundImage ="background.jpg"
             Title="Enter Location"
             >


    <ContentPage.Content>
        <StackLayout VerticalOptions="Center">
            <StackLayout Orientation="Horizontal" HorizontalOptions="Center" Margin="20">
                <Label Text="Welcome " FontSize="40" TextColor="White" FontAttributes="Bold"></Label>
                <Label Text="{Binding LoggedUser.Name}" FontSize="40" TextColor="White" FontAttributes="Bold" ></Label>
            </StackLayout>

            <Image Source="Logo.png" HeightRequest="100" WidthRequest="100"></Image>
            <Label Text="Enter Weather Location" FontSize="32" Padding="20" HorizontalOptions="CenterAndExpand" HorizontalTextAlignment="Center"
              TextColor="White"></Label>
            <StackLayout Padding="30" >
                <Entry x:Name="loc" Placeholder="Enter Location: " TextColor="White" PlaceholderColor="LightGray" FontSize="Large"></Entry>
            </StackLayout>
            <Button Clicked ="SubmitLocation" Text="Show Weather" Padding="20,10" BackgroundColor="Black" TextColor="White" BorderColor="White"
              WidthRequest="330" HorizontalOptions="Center" CornerRadius="20"></Button>
        </StackLayout>
    </ContentPage.Content>
</ContentPage>
```

Kod u C#

```csharp
using ProjekatRMA.Models;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using Xamarin.Forms;
using Xamarin.Forms.Xaml;

namespace ProjekatRMA.Views
{
    [XamlCompilation(XamlCompilationOptions.Compile)]
    public partial class WeatherLocation : ContentPage
    {
        public UserObj LoggedUser { get; set; }
        public WeatherLocation(UserObj dTO)
        {
            LoggedUser = dTO;
            BindingContext = this;
            InitializeComponent();
        }

        private void SubmitLocation(object sender, EventArgs e)
        {
            string location = loc.Text;
            Application.Current.MainPage.Navigation.PushAsync(new CurrentWeatherPage(location, LoggedUser));
        }
    }
}
```

6. Informacije o vremenu

Na ovoj stranici si prikazani podaci vremenske prognoze za uneseni grad. Podaci koji se prikazuju su: naziv grada, temperatura u celzijusima, vlažnost, brzina vjetra, pritisak i oblačnost. Na dnu stranice se nalazi vremenska prognoza za naredna četiri dana. Podaci se dobijaju eksterno koristeci OpenWeatherAPI.org.



Slika 5. Informacije o vremenu

Kod za CurrentWeatherPage.xaml

```
<?xml version="1.0" encoding="utf-8" ?>
<ContentPage xmlns="http://xamarin.com/schemas/2014/forms"
             xmlns:x="http://schemas.microsoft.com/winfx/2009/xaml"
             x:Class="ProjekatRMA.Views.CurrentWeatherPage"
             Title="Weather Info"
             >
    <Grid HorizontalOptions="FillAndExpand" VerticalOptions="FillAndExpand" RowSpacing="0">
        <Grid.RowDefinitions>
            <RowDefinition Height="*"/>
            <RowDefinition Height="Auto"/>
        </Grid.RowDefinitions>
        <Grid RowSpacing="0">
            <Grid.RowDefinitions>
                <RowDefinition Height="Auto"/>
                <RowDefinition Height="*"/>
            </Grid.RowDefinitions>
            <BoxView Grid.RowSpan="2" BackgroundColor="#7585BA" HorizontalOptions="FillAndExpand" VerticalOptions="FillAndExpand"/>
            <Image x:Name="bgImg" Aspect="AspectFill" Grid.RowSpan="2" Source="bg.png" Opacity="0.5" HorizontalOptions="FillAndExpand"
              VerticalOptions="FillAndExpand"/>
            <Image Aspect="AspectFill" Grid.RowSpan="2" Source="overlay.png" HorizontalOptions="FillAndExpand" VerticalOptions="FillAndExpand"/>
            <ImageButton Source="menu.png" Margin="20" WidthRequest="23" HeightRequest="15" HorizontalOptions="Start" VerticalOptions="Center"
              Clicked="TryAgain" />
            <Grid Grid.Row="1">
                <StackLayout Spacing="20">
                    <StackLayout Orientation="Horizontal" Spacing="50" HorizontalOptions="Center">
                        <StackLayout VerticalOptions="Center">
                            <Image x:Name="iconImg" Source="cloud.png" WidthRequest="67" HeightRequest="50"/>
                            <Label x:Name="descriptionTxt" Text="Cloudy" TextColor="White" FontSize="13" HorizontalOptions="Center"/>
                        </StackLayout>
                        <StackLayout VerticalOptions="Center">
                            <Label x:Name="cityTxt" Text="LONDON" TextColor="White" FontSize="20" FontAttributes="Bold" HorizontalOptions="Center"/>
                            <Label x:Name="dateTxt" Text="SATURDAY, NOV 30" TextColor="White" FontSize="13" HorizontalOptions="Center"/>
                        </StackLayout>
                    </StackLayout>
```

```
    </StackLayout>
    <StackLayout>
        <StackLayout Orientation="Horizontal" HorizontalOptions="Center" Spacing="0">
            <Label x:Name="temperatureTxt" Text="25" TextColor="White" FontSize="150" HorizontalOptions="Center"/>
            <Label Text="°" TextColor="White" FontSize="150" HorizontalOptions="Center"/>
        </StackLayout>
        <Label Text="CELSIUS" Margin="0,-30,0,0" TextColor="White" FontSize="13" HorizontalOptions="Center"/>
    </StackLayout>
    <Grid WidthRequest="320" ColumnSpacing="10" HorizontalOptions="CenterAndExpand" VerticalOptions="CenterAndExpand">
        <Grid.ColumnDefinitions>
            <ColumnDefinition Width="*"/>
            <ColumnDefinition Width="*"/>
            <ColumnDefinition Width="*"/>
            <ColumnDefinition Width="*"/>
        </Grid.ColumnDefinitions>
        <StackLayout Orientation="Vertical" Spacing="10" HorizontalOptions="CenterAndExpand">
            <Image Source="humidity.png" HeightRequest="25" HorizontalOptions="Center"/>
            <StackLayout Spacing="7" HorizontalOptions="CenterAndExpand">
                <Label x:Name="humidityTxt" Text="50%" TextColor="White" FontSize="14" FontAttributes="Bold" HorizontalOptions="Center"/>
                <Label Text="Humidity" TextColor="White" Margin="0,-5,0,0" FontSize="11" HorizontalOptions="Center"/>
            </StackLayout>
        </StackLayout>
        <StackLayout Grid.Column="1" Orientation="Vertical" Spacing="10" HorizontalOptions="CenterAndExpand">
            <Image Source="wind.png" HeightRequest="25" HorizontalOptions="Center"/>
            <StackLayout Spacing="7" HorizontalOptions="CenterAndExpand">
                <Label x:Name="windTxt" Text="2.6 m/s" TextColor="White" FontSize="14" FontAttributes="Bold" HorizontalOptions="Center"/>
                <Label Text="Wind" TextColor="White" Margin="0,-5,0,0" FontSize="11" HorizontalOptions="Center"/>
            </StackLayout>
        </StackLayout>
```

```
                <StackLayout Grid.Column="2" Orientation="Vertical" Spacing="10" HorizontalOptions="CenterAndExpand">
                    <Image Source="gauge.png" HeightRequest="25" HorizontalOptions="Center"/>
                    <StackLayout Spacing="7" HorizontalOptions="CenterAndExpand">
                        <Label x:Name="pressureTxt" Text="1011 hpa" TextColor="White" FontSize="14" FontAttributes="Bold" HorizontalOptions="Center"/>
                        <Label Text="Pressure" TextColor="White" Margin="0,-5,0,0" FontSize="11" HorizontalOptions="Center"/>
                    </StackLayout>
                </StackLayout>
                <StackLayout Grid.Column="3" Orientation="Vertical" Spacing="10" HorizontalOptions="CenterAndExpand">
                    <Image Source="cloudiness.png" HeightRequest="25" HorizontalOptions="Center"/>
                    <StackLayout Spacing="7" HorizontalOptions="CenterAndExpand">
                        <Label x:Name="cloudinessTxt" Text="24%" TextColor="White" FontSize="14" FontAttributes="Bold" HorizontalOptions="Center"/>
                        <Label Text="Cloudiness" TextColor="White" Margin="0,-5,0,0" FontSize="11" HorizontalOptions="Center"/>
                    </StackLayout>
                </StackLayout>
            </Grid>
        </StackLayout>
    </Grid>
</Grid>
<Grid Grid.Row="1" HeightRequest="160" ColumnSpacing="0">
    <Grid.ColumnDefinitions>
        <ColumnDefinition Width="*"/>
        <ColumnDefinition Width="*"/>
        <ColumnDefinition Width="*"/>
        <ColumnDefinition Width="*"/>
    </Grid.ColumnDefinitions>
    <Grid BackgroundColor="#758ABA" Opacity="0.4">
        <StackLayout HorizontalOptions="CenterAndExpand" VerticalOptions="CenterAndExpand">
            <Label x:Name="dayOneTxt" Text="Sunday" TextColor="White" FontSize="13" HorizontalOptions="Center"/>
            <Label x:Name="dateOneTxt" Margin="0,-5,0,0" Text="01 Dec" TextColor="White" FontSize="10" HorizontalOptions="Center"/>
            <Image x:Name="iconOneImg" Source="cloud.png" Margin="0,20" WidthRequest="30" HeightRequest="22"/>
            <StackLayout Orientation="Horizontal" HorizontalOptions="Center" Spacing="0">
                <Label x:Name="tempOneTxt" Text="22" TextColor="White" FontSize="23" HorizontalOptions="Center"/>
                <Label Text="°" TextColor="White" FontSize="23" HorizontalOptions="Center"/>
            </StackLayout>
        </StackLayout>
    </Grid>
```

```
            </Grid>
            <Grid BackgroundColor="#758ABA" Opacity="0.6" Grid.Column="1">
                <StackLayout HorizontalOptions="CenterAndExpand" VerticalOptions="CenterAndExpand">
                    <Label x:Name="dayTwoTxt" Text="Monday" TextColor="White" FontSize="13" HorizontalOptions="Center"/>
                    <Label x:Name="dateTwoTxt" Margin="0,-5,0,0" Text="02 Dec" TextColor="White" FontSize="10" HorizontalOptions="Center"/>
                    <Image x:Name="iconTwoImg" Source="cloud.png" Margin="0,20" WidthRequest="30" HeightRequest="22"/>
                    <StackLayout Orientation="Horizontal" HorizontalOptions="Center" Spacing="0">
                        <Label x:Name="tempTwoTxt" Text="17" TextColor="White" FontSize="23" HorizontalOptions="Center"/>
                        <Label Text="°" TextColor="White" FontSize="23" HorizontalOptions="Center"/>
                    </StackLayout>
                </StackLayout>
            </Grid>
            <Grid BackgroundColor="#758ABA" Opacity="0.8" Grid.Column="2">
                <StackLayout HorizontalOptions="CenterAndExpand" VerticalOptions="CenterAndExpand">
                    <Label x:Name="dayThreeTxt" Text="Tuesday" TextColor="White" FontSize="13" HorizontalOptions="Center"/>
                    <Label x:Name="dateThreeTxt" Margin="0,-5,0,0" Text="03 Dec" TextColor="White" FontSize="10" HorizontalOptions="Center"/>
                    <Image x:Name="iconThreeImg" Source="cloud.png" Margin="0,20" WidthRequest="30" HeightRequest="22"/>
                    <StackLayout Orientation="Horizontal" HorizontalOptions="Center" Spacing="0">
                        <Label x:Name="tempThreeTxt" Text="28" TextColor="White" FontSize="23" HorizontalOptions="Center"/>
                        <Label Text="°" TextColor="White" FontSize="23" HorizontalOptions="Center"/>
                    </StackLayout>
                </StackLayout>
            </Grid>
            <Grid BackgroundColor="#758ABA" Grid.Column="3">
                <StackLayout HorizontalOptions="CenterAndExpand" VerticalOptions="CenterAndExpand">
                    <Label x:Name="dayFourTxt" Text="Wednesday" TextColor="White" FontSize="13" HorizontalOptions="Center"/>
                    <Label x:Name="dateFourTxt" Margin="0,-5,0,0" Text="04 Dec" TextColor="White" FontSize="10" HorizontalOptions="Center"/>
                    <Image x:Name="iconFourImg" Source="cloud.png" Margin="0,20" WidthRequest="30" HeightRequest="22"/>
                    <StackLayout Orientation="Horizontal" HorizontalOptions="Center" Spacing="0">
                        <Label x:Name="tempFourTxt" Text="25" TextColor="White" FontSize="23" HorizontalOptions="Center"/>
                        <Label Text="°" TextColor="White" FontSize="23" HorizontalOptions="Center"/>
                    </StackLayout>
                </StackLayout>
            </Grid>
        </Grid>
    </Grid>
</ContentPage>
```

Kod u C#

```
using Newtonsoft.Json;
using ProjekatRMA.Helpers;
using ProjekatRMA.Models;
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

using Xamarin.Forms;
using Xamarin.Forms.Xaml;

namespace ProjekatRMA.Views
{

    [XamlCompilation(XamlCompilationOptions.Compile)]
    public partial class CurrentWeatherPage : ContentPage
    {
        public UserObj loggedUser { get; set; }
        public string Location { get; set; }
        public CurrentWeatherPage(string location, UserObj user)
        {
            loggedUser = user;
            Location = location;
            InitializeComponent();
            GetWeatherInfo();
        }


    private async void GetWeatherInfo()
    {
        var url = $"https://api.openweathermap.org/data/2.5/weather?q={Location}&appid=5ff982f884a179b772f1f65cbae537d6&units=metric";

        var result = await ApiCaller.Get(url);

        if (result.Successful)
        {
            try
            {
                var weatherInfo = JsonConvert.DeserializeObject<WeatherInfo>(result.Response);
                descriptionTxt.Text = weatherInfo.weather[0].description.ToUpper();
                iconImg.Source = $"w{weatherInfo.weather[0].icon}";
                cityTxt.Text = weatherInfo.name.ToUpper();
                temperatureTxt.Text = weatherInfo.main.temp.ToString("0");
                humidityTxt.Text = $"{weatherInfo.main.humidity}%";
                pressureTxt.Text = $"{weatherInfo.main.pressure} hpa";
                windTxt.Text = $"{weatherInfo.wind.speed} m/s";
                cloudinessTxt.Text = $"{weatherInfo.clouds.all}%";

                var dt = new DateTime().ToUniversalTime().AddSeconds(weatherInfo.dt);
                dateTxt.Text = dt.ToString("dddd, MMM dd").ToUpper();

                GetForecast();
            }
            catch (Exception ex)
            {
                await DisplayAlert("Weather Info", ex.Message, "OK");
            }
        }
        else
        {
            await DisplayAlert("Weather Info", "No weather information found", "OK");
        }
    }

```

```
66          private async void GetForecast()
67          {
68              var url = $"https://api.openweathermap.org/data/2.5/forecast?q={Location}&appid=5ff982f884a179b772f1f65cbae537d6&units=metric";
69              var result = await ApiCaller.Get(url);
70
71              if (result.Successful)
72              {
73                  try
74                  {
75                      var forcastInfo = JsonConvert.DeserializeObject<ForecastInfo>(result.Response);
76
77                      List<List> allList = new List<List>();
78
79                      foreach (var list in forcastInfo.list)
80                      {
81                          //var date = DateTime.ParseExact(list.dt_txt, "yyyy-MM-dd hh:mm:ss", CultureInfo.InvariantCulture);
82                          var date = DateTime.Parse(list.dt_txt);
83
84                          if (date > DateTime.Now && date.Hour == 0 && date.Minute == 0 && date.Second == 0)
85                              allList.Add(list);
86                      }
87
88                      dayOneTxt.Text = DateTime.Parse(allList[0].dt_txt).ToString("dddd");
89                      dateOneTxt.Text = DateTime.Parse(allList[0].dt_txt).ToString("dd MMM");
90                      iconOneImg.Source = $"w{allList[0].weather[0].icon}";
91                      tempOneTxt.Text = allList[0].main.temp.ToString("0");
92
93                      dayTwoTxt.Text = DateTime.Parse(allList[1].dt_txt).ToString("dddd");
94                      dateTwoTxt.Text = DateTime.Parse(allList[1].dt_txt).ToString("dd MMM");
95                      iconTwoImg.Source = $"w{allList[1].weather[0].icon}";
96                      tempTwoTxt.Text = allList[1].main.temp.ToString("0");
97
98                      dayThreeTxt.Text = DateTime.Parse(allList[2].dt_txt).ToString("dddd");
99                      dateThreeTxt.Text = DateTime.Parse(allList[2].dt_txt).ToString("dd MMM");
100                     iconThreeImg.Source = $"w{allList[2].weather[0].icon}";
101                     tempThreeTxt.Text = allList[2].main.temp.ToString("0");
102
102
103                         dayFourTxt.Text = DateTime.Parse(allList[3].dt_txt).ToString("dddd");
104                         dateFourTxt.Text = DateTime.Parse(allList[3].dt_txt).ToString("dd MMM");
105                         iconFourImg.Source = $"w{allList[3].weather[0].icon}";
106                         tempFourTxt.Text = allList[3].main.temp.ToString("0");
107
108                 }
109                 catch (Exception ex)
110                 {
111                     await DisplayAlert("Weather Info", ex.Message, "OK");
112                 }
113             }
114             else
115             {
116                 await DisplayAlert("Weather Info", "No forecast information found", "OK");
117             }
118         }
119
120         private void TryAgain(object sender, EventArgs e)
121         {
122             Application.Current.MainPage.Navigation.PushAsync(new WeatherLocation(loggedUser));
123         }
124     }
125 }
```

7. Zaključak

U današnjem vremenu skoro je nemoguće adekvatno izvršiti obaveze van kuće bez informacija o vremenu. Zbog klimatskih promjena ne možemo biti sigurni kakvo vrijeme možemo očekivati. Adekvatno rješenje ovog problema je aplikacija vremenske prognoze. Danas svi posjeduju mobilne uređaje i nose ih sa sobom gdje god da krenu. Pomoću ove aplikacije brzo su nam dostupne sve potrebne informacije za izlazak vani.

Aplikacija je dizajnirana da nakon otvoranja traži prijavu. Nakon toga sve što se treba uraditi je upisati ime grada o kojem želimo dobiti informacije. Prikaz informacija je jednostavan i svima razumljiv. Prikazuje se naziv grada, temperatura u stepenima celzija, vlažnost, oblačnost, pritisak, brzina vjetra te temepratura za nardna četiri dana.

U ovom radu je prikazan izgled aplikacije kroz slike (1-5), pored izgleda kroz slike je prstavljen i kod iz .xaml i .xaml.cs stranica.

8. Popis slika

Slika 1. Početna stranica aplikacije..... 2

Slika 2. Stranica za registraciju..... 5

Slika 3. Login..... 8

Slika 4. Stranica za unos lokacije..... 10

Slika 5. Informacije o vremenu..... 12